Adaptive Power Gating for Function Units in a Microprocessor

Kimiyoshi Usami[1], Tatsunori Hashida[1], Satoshi Koyama[1], Tatsuya Yamamoto[1], Daisuke Ikebuchi[2], Hideharu Amano[2], Mitaro Namiki[3], Masaaki Kondo[4], Hiroshi Nakamura[5]
[1]Shibaura Institute of Technology, 3-7-5 Toyosu, Kohtoh-ku, Tokyo, Japan
[2]Keio University, 3-14-1 Hiyoshi, Kohoku-ku, Yokohama, Japan
[3]Tokyo University of Agriculture and Technology, 2-24-16 Naka-cho, Koganei-shi, Tokyo, Japan
[4]The University of Electro-Communications, 1-5-1 Chofugaoka, Chofu-shi, Tokyo, Japan
[5]The University of Tokyo, 4-6-1 Komaba, Meguro-ku, Tokyo, Japan
[1]E-mail: usami@shibaura-it.ac.jp

Abstract

This paper describes adaptive fine-grain control to power gate function units based on temperature dependent break-even time (BET). An analytical model to express the temperature dependent BET is introduced and the accuracy of the model was examined. Results demonstrated that the model well represents the exponential decrease in BET with the temperature. Meanwhile, it was found that the accuracy gets worse at higher temperature and the cause is energy dissipation due to transient glitch at the wakeup. We propose four power-gating policies employing time-based or history-based approaches. Effectiveness in energy savings was evaluated using real design data of four function units in a microprocessor implemented in a 65nm technology. Results showed that introducing adaptive control to make use of temperature-dependent BET enhances energy savings by up to 21% in the time-based approach and by up to 18% in the history-based approach. The adaptive history-based policy with a limiter outperforms the adaptive time-based policy in energy savings and reduces the total energy of four function units to 11.8% at 100°C as compared to the non-power-gating case.

Keywords

Power gating, adaptive, temperature, leakage, function unit

1. Introduction

Leakage power dissipation is one of the major concerns in microprocessors that use deeply scaled devices. Among existing techniques to reduce leakage power, power gating (PG) is one of the most promising approaches. Power gating is a technique to shut off the power supply to a circuit by turning off an embedded power-switch (PS) transistor during periods of inactivity. In commercial microprocessors or SoCs, power gating control has been mainly implemented in a coarse granularity so far. IP cores such as CPU or DSP cores in a SoC are power gated and unused cores are dynamically powered off [1][2].

In contrast, more aggressive techniques to power gate internal circuits in finer granularity at run time have been studied. In [3], the authors presented a technique to power gate execution units such as a fix-point unit and a floating-point unit in a microprocessor. In [4], an approach has been proposed to power gate function units such as an ALU, a shifter, a multiplier or a divider in a CPU core by pre-decoding instructions.

In these fine-grain power gating (FGPG) approaches, energy saving efficiency becomes more sensitive to energy overhead caused by powering off/on the circuits, as compared to coarse-grain techniques. To gain in energy savings, the powered-off period has to be long enough to compensate the energy overhead. There are two keys in FGPG: one is the break-even time (BET) and the other is how to control PG based on BET. BET is the time at which leakage energy saved by powering off becomes equal to the energy overhead. Hu et al proposed an analytical model for BET [3] and the model has been used in successive studies [5] [6]. However, how the model should be modified for temperature variations are not presented in those papers. Normally the chip temperature goes up during the operation due to its own heat dissipation. In [7], the authors reported simulation results showing that BET for function units significantly decreases as the temperature is elevated. However, it was not addressed how the analytical model would be affected by the temperature variations.

How to control PG based on BET is another important issue. Ideally, PG should be enabled only at sleep events whose idle time is longer than BET. PG should be disabled at sleep events whose idleness is shorter than BET. However, at the time when the circuit becomes idle, it is impossible to precisely predict whether the idleness continues over BET. In [3], the authors proposed a Time-Based (TB) policy to count idle cycles from the beginning of idleness and to enable PG if the counted value reaches BET. This approach has an advantage that it suppresses powering off at a very short idleness that fails to gain in energy savings. Meanwhile, this policy has a drawback that it misses potential leakage savings during the period in which idle cycles are counted. In addition to the TB policy, several PG policies have been proposed so far. Compiler-based approaches were studied in [8]. An approach to change the threshold to power off the function unit depending on applications was studied in [9]. However, an idea to proactively make use of the temperature dependence of BET has not been presented. We conducted a preliminary study on temperature-dependent BET modeling for a multiplier in which internal arrays are power gated in fine granularity [10]. However, validity of the derivation process for analytical expressions at high temperature was not examined in detail. Causes of the difference between the

BET value obtained by the model and that by experiments were not investigated, either.

In this paper, we present an analytical model to express the temperature dependent BET and propose power-gating policies to make use of the temperature dependent BET for energy savings. The contributions of this paper are three-fold:

(1) We re-examined the Hu's analytical model for BET and proved that the expressions keep validity at the high temperature. Accuracy of the derived model for the temperature dependent BET was examined and the cause of the difference between the model and the experimental value was identified as the transient-glitch energy.

(2) We proposed four power-gating policies and demonstrated that introducing adaptive control enhances energy savings at high temperatures. In particular, we found that an adaptive history-based policy with a limiter outperforms other approaches and achieves very close energy savings to those in the oracle case.

(3) As an evaluation platform, we designed and implemented a MIPS R3000 based CPU chip in which function units such as ALU, a shifter, a multiplier and a divider are power gated. All evaluations were conducted based on the data from the layout of this design.

The rest of the paper is organized as follows: Section 2 describes an analytical model for the temperature dependent BET. Section 3 presents power-gating policies that we propose. Section 4 describes an experimental setup for evaluation and Section 5 presents the results.

2. Temperature-dependent model for BET

Hu *et al* [3] derived the following expression for the break-even cycles BEC for obtaining energy savings in power gating under an assumption that a pMOS header PS is used:

$$BEC = 2\frac{1}{L\alpha}\sqrt{\frac{mU_tW_H}{V_{dd}\eta}(1+2\frac{C_D}{C_S})} \qquad (1)$$

where L is the leakage factor defined as a ratio of the average leakage energy in the active state and the average switching energy; α is the switching activity averaged over the whole circuit; m is the body-effect coefficient [13]; U_t is the thermal voltage; W_H is the ratio of the total area of PS to the area of the power-gated circuit; V_{dd} is the supply voltage; η is the DIBL factor; C_D is the total capacitance at the virtual-V_{dd} line; and C_S is the total switching capacitance in the circuit. When idle cycles exceed BEC, turning PS gives energy savings because the total (aggregate) saved leakage-energy exceeds energy overhead for driving PS. In the derivation process for expression (1), however, no attention has been paid to the temperature at all. To derive an expression for temperature-dependent BEC, it is required to re-check if the derivation steps still work when the temperature changes. We found that approximations employed in the derivation cannot be applied as they are when leakage current increases drastically at high temperature. To overcome this problem, we re-built the derivation process such that approximations can be applied even at high temperature. Details are presented in Appendix.

As a result, we reached exactly the same expression for BEC as (1) for high temperature. This means that expression (1) works irrespective of temperature. We further confirmed that BEC can be expressed by equation (1) for a power-gated circuit with nMOS footer PS as well as that with pMOS header PS.

The break-even time BET can be expressed as

$$BET = BEC \times T_{cyc} = 2\frac{T_{cyc}}{L\alpha}\sqrt{\frac{mU_tW_H}{V_{dd}\eta}(1+2\frac{C_D}{C_S})} \qquad (2)$$

where T_{cyc} is the cycle time. Next we would like to look at the temperature dependence of BET. In expression (2), the thermal voltage U_t defined as kT/q is an explicit temperature-dependent parameter, where k is the Boltzmann's constant; and q is the electronic charge. In addition, L is temperature dependent because L depends on leakage energy. It should be noted that the major contributor to the leakage current is assumed to be subthreshold leakage in this model. Subthreshold leakage current can be written as

$$I_L(T) = \mu_0 C_{OX}\frac{W_{MOS}}{L_{MOS}}(m-1)U_t^{\,2}\exp(-\frac{V_T}{mU_t}) \qquad (3)$$

where μ_0 is the mobility; C_{OX} is the oxide capacitance of a MOSFET; L_{MOS} is the MOSFET channel length; W_{MOS} is the total width of MOSFETs in the circuit; and V_T is the threshold voltage of a MOSFET in power gated circuit. The leakage factor L can be expressed as

$$L = \frac{T_{cyc}I_L(T)V_{dd}}{(1/2)\alpha C_S V_{dd}^{\,2}} = \frac{2T_{cyc}}{\alpha C_S V_{dd}}\mu_0 C_{OX}\frac{W_{MOS}}{L_{MOS}}(m-1)U_t^2\exp(-\frac{V_T}{mU_t}). \qquad (4)$$

By substituting (4) for (2), we have

$$BET = K_1 T^{-\frac{3}{2}}\exp(\frac{K_2}{T}) \qquad (5)$$

where

$$K_1 = \frac{C_S V_{dd}^{1/2}\cdot(k/q)^{-3/2}}{\mu_0 C_{OX}\cdot(W_{MOS}/L_{MOS})(m-1)}(\frac{mW_H}{\eta})^{1/2}(1+2\frac{C_D}{C_S})^{1/2}$$

and $K_2 = qV_T/mk$.

We define *break-even time ratio BETR* as the ratio of the BET at the temperature T over the BET at the room temperature T_{RT} (i.e. 25°C or 298K). *BETR* at the temperature T can be expressed as

$$BETR(T) = T_{RT}^{\frac{3}{2}}\exp(-\frac{K_2}{T_{RT}})\,T^{-\frac{3}{2}}\exp(\frac{K_2}{T}). \qquad (6)$$

For V_T = 0.4V and m=1.38, $K_2 \approx 3360$. Notice that BETR reduces almost exponentially with the temperature. As compared to the room temperature, BET reduces to 22% at 65°C and to 7% at 100°C.

3. Proposed power-gating policies

We describe four PG policies that we propose. In this section we use the term BEC (break-even cycles) instead of BET because PG is controlled while detecting the number of idle cycles.

3.1. Adaptive Time-Based (ATB) policy

The first approach is an Adaptive Time-Based (ATB) policy. In this approach, the chip temperature is monitored using an on-chip temperature sensor, and PG control is

adaptively changed based on the BEC value at the detected temperature. In the original time-based (TB) policy [3], PG is controlled based on one fixed BEC value. In other words, even though the chip temperature is changed, enabling/disabling PG is conducted based on one BEC (typically BEC at the room temperature). As described in Section 2, BET (and hence BEC as well) reduces exponentially with the temperature. This means that counting idle cycles until the room-temperature's BEC further misses potential leakage-savings at higher temperature. The ATB approach overcomes this drawback by introducing a temperature sensor and adaptively controlling PG based on the temperature-dependent BEC. Since temperature increases are gradual, we assume that appreciable temperature changes occur only at 10,000 cycle granularity [11]. We read the on-chip temperature sensor at this granularity, refer to a temperature-BEC translation table, and store the obtained BEC value into a BEC register. This updating process for the BEC register is done only when the temperature is monitored. Time-based control is performed based on the data in the BEC register. It should be noted that BECs at various temperatures need to be characterized and recorded in the translation table in advance.

Equipping with on-chip temperature sensors has become popular in modern high-performance microprocessors, mainly for protecting the chip from overheating. In [12], the authors presented a digital temperature sensor circuit consisting of a ring oscillator and a counter. The frequency of the ring oscillator is controlled by a temperature-sensitive current reference, and the counter records the number of oscillations within set time interval. A digital temperature sensor like this can be employed in the ATB approach. Since the temperature sensor can be powered off at any other time but temperature detection, introducing a temperature sensor would not give a big impact on the entire energy dissipation.

3.2. History-Based (HB) policy

The second approach that we propose is a History-Based (HB) policy. In this policy, we predict whether the idle cycles of the next sleep event will exceed BEC or not, based on the idle cycles of the preceding sleep event. In other words, if the idle cycles of the preceding sleep event (i.e. the sleep event just before) exceeded BEC, we predict that the idle cycles of the next sleep event will exceed. Based on this prediction, we turn off PS immediately when the idleness is detected. Inversely, if the idle cycles of the preceding sleep event did not exceed BEC, we predict that the next idleness will be short and hence disable PG.

At any sleep event, sleep cycles are counted by a counter and the total idle cycles for the sleep event are recorded in a register. The stored data in the register is compared with BEC, and if it exceeds BEC then we set a prediction-flag register to '1'. Otherwise, we reset the flag register to '0'. Recording a history of previous operations and employing it for predictions are quite popular in the computer architecture arena. A branch prediction is one of the most well-known techniques. This technique is to look up the address of the instruction to see if a branch was taken the last time this instruction was executed, and if so, to begin fetching new instructions from the same place as the last time. In contrast to the branch prediction, in the HB approach we do not record the address of instructions. Instead, we only record the information on whether the idle cycles at the preceding sleep event exceeded BEC or not.

As compared to the TB policy, the HB approach has an advantage that PS can be turned off immediately after the idleness is detected. This allows us to overcome the weakness of the TB policy that misses potential leakage savings during the period in which idle cycles are counted. Instead, effectiveness of the HB policy depends on whether the prediction hits or not. This will be examined and discussed in Section 5.

Required hardware components in the HB policy are the same as those in the TB policy except a prediction-flag register (i.e. 1-bit flip-flop). Both policies require an idle-cycle counter, a register to record the total idle cycles at a sleep event, a register to record BEC, and a comparator to compare the total idle cycles at a sleep event with BEC.

Notice that in the HB policy, the BEC value at the room temperature is used at all times.

3.3. Adaptive History-Based (AHB) policy

The third approach that we propose is an Adaptive History-Based (AHB) policy. In this policy, enabling/disabling PG is determined based on the history of the preceding sleep event, like the HB policy, but the prediction is done through a comparison with the temperature dependent BEC. Like the ATB approach, the temperature is monitored using an on-chip temperature sensor and BEC at the detected temperature is recorded in a BEC register. The idle cycles of the preceding sleep event are compared with this BEC data and the result is recorded in a prediction-flag register. Since BEC reduces as the temperature is elevated, it becomes more likely that the number of idle cycles at a sleep event exceed BEC at higher temperature. This results in increasing opportunities to enable PG at higher temperatures. The BEC register is updated only when the temperature monitoring is done. As compared to the HB policy, the AHB approach requires an on-chip temperature sensor as an extra hardware component.

3.4. Adaptive History-Based policy with limiter (AHB-LM)

The forth approach we propose is an Adaptive History-Based policy with limiter (AHB-LM). In the AHB policy described above, if the prediction is a "short sleep", PG is disabled at the next sleep event. However, despite the prediction, if the sleep is actually long, leakage keeps flowing and no energy is saved. To minimize this wasted energy, we introduce a time limiter to the AHB policy. If the prediction of short-sleep misses, we count the clock cycles after idleness is detected. If the counted cycles reach BEC, we turn off PS to avoid further wasting of leakage energy.

Notice that no extra hardware is needed to introduce a limiter to AHB. This is because in the AHB policy a counter to count the clock cycles after the idleness detection is already equipped. In AHB, the counter measures the sleep period at every sleep event and the obtained sleep period is compared with BEC for the next prediction using a

comparator. The counter and the comparator can be shared in AHB-LM.

4. Setup for experiment

As an evaluation platform, we designed a MIPS R3000 based CPU core, *Geyser-1*, whose function units (ALU, shifter, multiplier, and divider) are dynamically power gated. We implemented a chip in a commercial 65nm technology and conducted evaluations using data extracted from the layout.

4.1. Control scheme for power gating

Power gating is controlled at the instruction-by-instruction basis in Geyser-1. Function units to be activated are detected in a processor pipeline. The pipeline is a standard five-stage structure consisting of Instruction Fetch (IF), Instruction Decode (ID), Execution (EX), Memory Access (MEM) and Write Back (WB) stages. In Geyser-1, a sleep policy to *put a function unit to sleep whenever it is idle* was adopted as a default. We refer to this policy as WIPS (Whenever Idle Put to Sleep). Function units except ALU enter the power-off state immediately after finishing the operation. Sleep signals for target units are generated in an on-chip sleep controller. Function units to be activated could be detected at the ID stage but it does not meet the timing to start execution at the EX stage, because a time is required to wakeup a unit in the power-off state. Using an implementation scheme described in Section 4.2, we achieved a wakeup time less than a cycle at the target operation frequency 200MHz. However, it still requires one cycle stall between the ID and EX stages, leading to performance degradation. To avoid this, we introduced a pre-wakeup approach [4] to detect a unit to be activated at the IF stage and wakeup the unit at the ID stage. A very simple decoder was implemented in the IF stage just to detect a unit to be activated.

Geyser-1 has a feature to disable power gating for each function unit. Control registers for this purpose are provided so as to allow the software (e.g. the operating system) to change the information in the register.

4.2. Implementation scheme

We implemented power gating for the function units using a Locally-Shared Virtual ground (LSV) scheme presented in [5]. In this scheme, the entire circuit for a function unit is partitioned into smaller local power domains. Logic gates in a local power domain are connected to an nMOS footer PS transistor via a local virtual-ground (VGND) line. Power switches are shared only within the local power domain. Although power switches are controlled by a unique sleep signal in a unit, the power-switch size is tuned independently in a local power domain. Furthermore, since the real ground rail still remains within power-gated cells in this scheme, non-power-gated cells such as flip-flops, clock buffers, repeaters, power-switch drivers (PSDs) and isolation cells are allowed to be placed at any location in a row.

To implement the LSV structure, we developed the following design flow. From RTL, we synthesize a netlist using low-Vth standard cell library. Isolation cells to avoid signal floating are inserted into the boundary between power-gated and non-power-gated portions. After performing timing-driven placement, we swap a logic cell with the same-sized power-gating cell (PG-cell). The PG-cell is a low-Vt logic cell with the same functionality as the original standard cell but has a VGND pin. The source of an nMOS transistor in the PG-cell is connected to a VGND pin instead of the real ground rail. After inserting properly sized power-switch cells and PSDs, local VGND lines are routed along with signal wires at the final routing step. A dedicated commercial CAD tool *CoolPower* [14] is used at this step. To suppress rush current at the wakeup, PSDs are sized such that the time to start powering-on each local domain are not overlapped. Flip-flops are out of our target to power gate because of performance and energy overhead for saving and restoring the data. Since PG-cells and non-power-gated cells can exist together in a row, we employ the conventional timing-driven P&R and clock-tree synthesis to generate the final layout.

Figure 1 shows the layout of the chip. Table 1 summarizes the results from applying PG to the function units. The overhead of both power switch and isolation cells was approximately 5% - 8% in cell count.

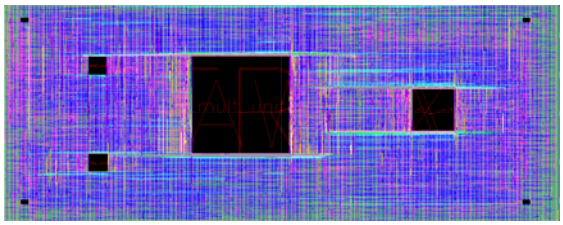

Figure 1: Layout of *Geyser-1* chip. Black boxes show power-gated function units.

Table 1: Implementation results of PG for function units.

	ALU	Shifter	Multiplier	Divider
# Logic Cells	1690	1628	9252	14837
# PS Cells	106	100	648	704
# Isolation Cells	33	32	64	64
# Total cells	1829	1760	9964	15605

4.3. Evaluation scheme

We conducted a post-layout circuit-level simulation using Hsim for each function unit to analyze energy dissipation during the sleep operation. At the evaluation of power-gating policies, a Verilog simulation for the entire RTL model of the CPU was employed as well. The Verilog simulation was executed while running two application programs: (i) a DCT (Discrete Cosine Transform) program used at multi-media processing, and (ii) a quick sort program (QSORT) in MiBench. To execute these programs at the Verilog simulations within a realistic time, we limited the input data size. For the DCT program we chose 16x16 DCT, whereas for the QSORT we assumed a sorting for 100 elements. The total clock cycles in the Verilog simulation for DCT were 43,232 cycles and those for QSORT were 188,569 cycles.

5. Experimental results

5.1. Evaluation for BETR

A parameter BETR was introduced in Section 2 to see the temperature dependence of BET. We compare the BETR value calculated from the analytical model and the values obtained experimentally from actual designs. Figure 2 shows the result.

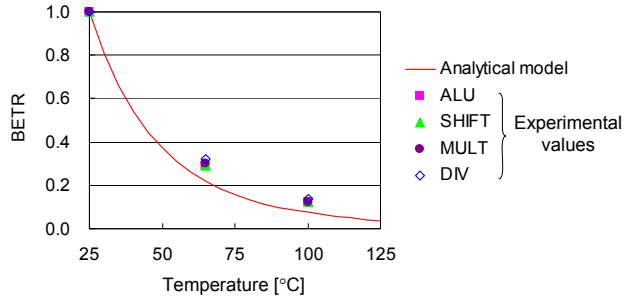

Figure 2: BETR obtained from the model and experiment.

The BETR values obtained from actual designs, denoted as *experimental values*, reduce to 0.30 at 65°C and to 0.13 at 100°C on average. It should be noted that those values for ALU, shifter, multiplier and divider are almost constant at individual temperatures despite different circuit structures. The fact that BETR does not depend on the circuit structure can be predicted by analytical expression (6). This fact has been confirmed at above experimental results.

Another important observation is that all experimental values are larger than the value obtained from the analytical model. It was found that the difference was 36% at 65°C and 86% at 100°C on average. To investigate the reason of this difference, we analyzed using a Hsim simulator how the energy is dissipated in power-gated units after PS is turned off. We define "turning-off the power switch" as *sleep-in* and "turning-on the power switch" as *wakeup*. The sleep period is defined as a consecutive period that starts at the sleep-in and ends at the wakeup. In the sleep period, energy is dissipated in three ways:

1. Leakage energy E_{Leak_sleep} dissipated by leakage current to charge up output nodes of power-gated circuits and VGND in the sleep period,
2. Switching energy E_{PS} dissipated at the re-buffering network for PS occurring at sleep-in and wakeup, and
3. Transient-glitch energy E_{Glitch} dissipated at the wakeup.

We obtained E_{Leak_sleep} by monitoring the current flowing from the supply into power-gated circuits during the period from the sleep-in through the time just before the wakeup. Regarding E_{PS}, we defined a different power supply to the re-buffering network for PS in the simulation and then monitored the current flowing into the re-buffering network both at the sleep-in cycle and the wakeup cycle. This is because in our design the switching at the re-buffering network is completed within a cycle. E_{Glitch} was obtained by monitoring the current flowing from the supply through the circuit which is just woken up. This monitoring is done only at the wakeup cycle, because the wakeup is completed within a cycle as described in Section 4.1. We analyzed E_{Leak_sleep}, E_{PS} and E_{Glitch}, while changing the length of sleep period. Results at 65°C for the multiplier are shown in Figure 3. Energy dissipation for non-power-gated, $E_{non\text{-}PG}$, is also depicted in the figure. The cross point of $E_{non\text{-}PG}$ and the total energy of E_{Leak_sleep}, E_{PS} and E_{Glitch} is the break-even point. The length of sleep time for the break-even point is the actual BET. If we exclude E_{Glitch} from the total energy dissipation, BET reduces from 180ns to 100ns at the multiplier. We performed similar analyses at other function units as well and calculated BETR when excluding E_{Glitch}. Results for 65°C and 100°C are summarized in Table 2(a) and (b), respectively.

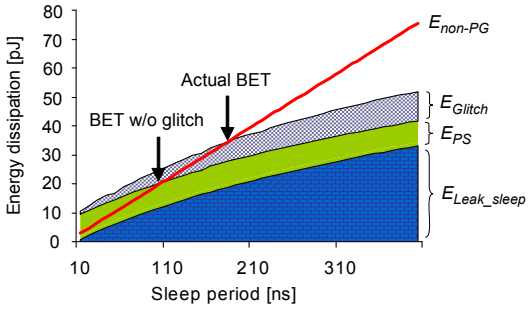

Figure 3: BET values for multiplier at 65°C.

Table 2(a): Comparisons for BETR at 65°C.

	Analytical model	Experimental values			
		Actual value		w/o E_{Glitch}	
		BETR	diff	BETR	diff
ALU	0.22	0.29	32%	0.25	14%
Shifter	0.22	0.29	32%	0.27	23%
Multiplier	0.22	0.30	36%	0.25	14%
Divider	0.22	0.32	45%	0.29	32%
average	0.22	0.30	36%	0.27	23%

Table 2(b): Comparisons for BETR at 100°C.

	Analytical model	Experimental values			
		Actual value		w/o E_{Glitch}	
		BETR	diff	BETR	diff
ALU	0.07	0.12	71%	0.09	29%
Shifter	0.07	0.12	71%	0.09	29%
Multiplier	0.07	0.12	71%	0.05	-29%
Divider	0.07	0.14	100%	0.10	43%
average	0.07	0.13	86%	0.08	14%

By excluding E_{Glitch} from the total energy dissipation, the difference between the analytical model and the experimental values reduces on average from 36% to 23% at 65°C, and from 86% to 14% at 100°C, respectively. This means that the difference between the actual values of BETR and the analytical model is resulted from the transient-glitch energy.

We analyzed the behavior of E_{Glitch} more in detail by changing the length of sleep period and the temperature. Results for the multiplier are shown in Figure 4. E_{Glitch} increases not only with the length of sleep period but also with the temperature. We think that this is because the glitch generation and propagation probabilities will get higher at longer sleep period or at higher temperature. At longer sleep period, more electric charge is accumulated into the

capacitance of output node of each gate. As a result, the voltage of the output node of each gate is raised closer to VDD and hence transient-glitch generation and propagation become easier. At higher temperature, increased leakage accelerates charging up the capacitance of the output nodes. The voltage of the output nodes is raised for a shorter period and hence transient-glitch generation and propagation become easier. In a circuit simulation to analyze node voltages, we actually observed the phenomenon that transient glitch becomes more likely at higher temperature.

As compared to the room temperature, energy overhead due to the transient-glitch energy becomes larger at higher temperature. Because of this fact, experimental values for BETR are larger than the value obtained from the analytical model.

Thus, it was found that the expression (6) well represents the temperature dependence of BET as the first-order approximation. On the other hand, it was also demonstrated that the inaccuracy of the analytical model is caused by transient-glitch energy.

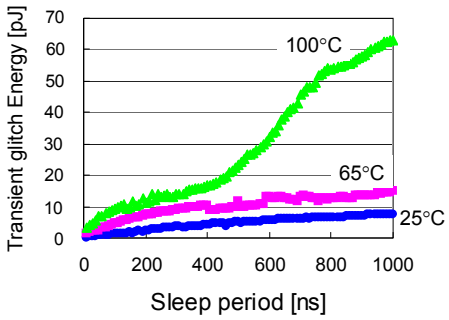

Figure 4: Transient-glitch energy as the function of the length of sleep period and the temperature at a multiplier.

Increase of E_{Glitch} shows a very irregular behavior as the temperature gets higher. This implies that the transient glitch has a very complicated mechanism in its generation and propagation. From this observation, rather than creating an analytical model for the transient-glitch energy, we chose to characterize energy dissipation during the sleep period for power-gated units by changing the length of sleep period and the temperature. Energy model obtained by this characterization is used in the evaluations below.

5.2. Evaluation for power-gating policies

We evaluated effectiveness of the proposed power-gating policies from the viewpoint of energy dissipation. As the target of comparison, we chose two conventional policies, i.e. the Time-Based (TB) policy and the Whenever-Idle-Put-to-Sleep (WIPS) policy described earlier. Four proposed policies (i.e. ATB, PB, APB and APB-TM) were compared with the conventional policies described above. Energy dissipations for four function units in Geyser-1 were analyzed assuming that each of those power-gating policies was applied to them. As a baseline, we took leakage energy of a circuit when PG is not conducted (denoted as "non-PG"). Energy dissipation when applying a power-gating policy was computed in the following way: first we characterized energy dissipation during the sleep period for each function unit by changing the length of sleep period and the temperature. Energy overhead occurring at the sleep-in and the wakeup is included in this energy dissipation. Next we conducted a Verilog simulation while running an application program to analyze the occurrence of consecutive idleness and the idle cycles for each function unit. Finally, by combining these results, we obtained energy dissipation. Figure 5 (a)-(f) show the results.

We first look at energy savings at the conventional policies, TB and WIPS. As shown in Figure 5 (a) and (d), the total energy of the function units was saved in TB for both DCT and QSORT programs. In contrast, the total energy is increased for DCT in the WIPS approach. To describe this reason, we define the following terms: *short-sleep* and *long-sleep*. A *short-sleep* is defined as a sleep event whose sleep period is shorter than BEC, while a *long-sleep* is defined as a sleep event whose sleep period is equal or longer than BEC. By analyzing the occurrence of *short-sleep* and *long-sleep* in DCT and QSORT programs, we found that *short-sleep* shares more than 99% in the total sleep events in DCT at ALU, a shifter and a multiplier. Since in WIPS the circuit is powered off at all sleep events, energy overhead due to sleep-in and wakeup increases the total energy. Next, we discuss effectiveness of introducing an adaptive control to a time-based policy by comparing TB and ATB. As for the total energy of the four function units, ATB saves more energy than TB by 5% at 65°C and by 17% at 100°C for DCT. For QSORT, ATB saves more energy than TB by 10% at 65°C and by 21% at 100°C. Among the function units, ATB saves more energy than TB at the multiplier and the shifter. Meanwhile, effectiveness of introducing an adaptive control does not effect at all at ALU. The reason of this can be explained from the occurrence of *short-sleep* and *long-sleep* in a program. Table 3 summarizes results from the occurrence analysis.

As shown in Table 3, *long-sleep* events exist for the shifter and the multiplier in the DCT and QSORT programs. In these *long-sleep* events, turning off PS as early as possible after the idleness detection minimizes wasting leakage energy. As compared to TB which uses the fixed BEC (i.e. BEC at 25°C) at all temperatures, ATB uses a shortened BEC at higher temperature. This enables ATB to turn off PS earlier than TB, resulting in reduction of wasted leakage energy. In the divider for QSORT, ATB reduces more energy than TB for the same reason. In contrast, for ALU, ATB did not reduce energy as well as TB. This is because the sleep period of all sleep events was still shorter than BEC in spite of reduction of BEC at high temperatures.

Effectiveness of introducing an adaptive control appears also in a history-based approach. As shown in Figure 5 (b) and (c), AHB saves more energy than HB by 7% at 65°C and by 18% at 100°C for DCT. In contrast to the time-based approach, in the history-based approach energy reduction is affected by the hit rate of the prediction.

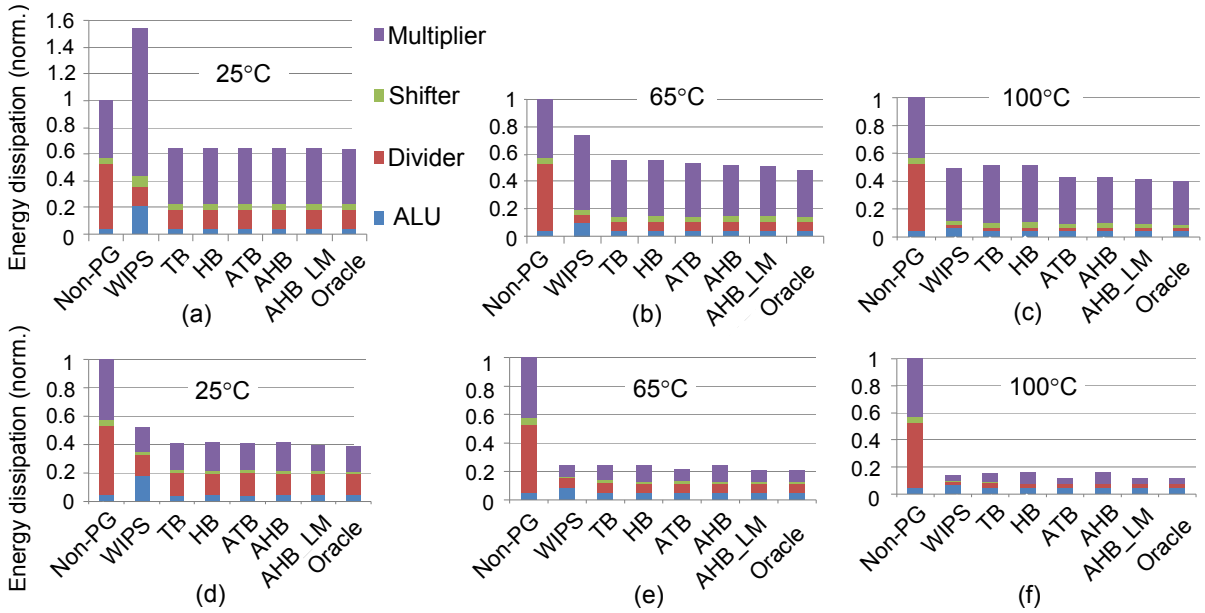

Figure 5: Energy dissipations for various power-gating policies; (a)-(c) for DCT, (d)-(f) for QSORT.

Table 3: Results from analysis for sleep events.

	ALU			Shifter			Multiplier			Divider		
Temperature [°C]	25	65	100	25	65	100	25	65	100	25	65	100
BEC	144	42	18	146	42	18	122	36	16	76	24	12
DCT												
Total sleep events	5532			1443			2544			1		
long_sleep	0	0	0	6	207	435	6	192	240	1	1	1
HIT Rate (long)	0%	0%	0%	0%	7%	23%	0%	49%	60%	0%	0%	0%
short_sleep	5532	5532	5532	1437	1236	1008	2538	2352	2304	0	0	0
HIT Rate (short)	100%	100%	100%	100%	84%	67%	100%	96%	96%	0%	0%	0%
QSORT												
Total sleep events	18937			234			108			40		
long_sleep	0	0	0	210	210	213	85	85	85	40	40	40
HIT Rate (long)	0%	0%	0%	90%	90%	90%	75%	75%	75%	98%	98%	98%
short_sleep	18937	18937	18937	24	24	21	23	23	23	0	0	0
HIT Rate (short)	100%	100%	100%	13%	13%	0%	13%	13%	13%	0%	0%	0%

We define the *hit rate for long-sleep* as the ratio of successful prediction in which the prediction is *long-sleep* and the actual sleep event is the same. The *hit rate for short-sleep* is defined in the same way. The *hit rate for long-sleep* and the *hit rate for short-sleep* are summarized in Table 3. In DCT, the *hit rate for long-sleep* of the multiplier gets higher as the temperature goes up. This results in more energy savings in AHB than HB. In contrast to DCT, for QSORT, AHB does not show any better energy savings over HB both at 65°C and 100°C. This is because the *hit rate for long-sleep* does not increase at the shifter, the multiplier and the divider even though the temperature is elevated.

We finally compare ATB and AHB in energy savings. In DCT, AHB saves more energy than ATB both at 65°C and at 100°C, whereas in QSORT, AHB saves less energy than ATB. We investigated factors that degrade energy savings of AHB. As shown in Table 3, in QSORT, the *hit rate for short-sleep* is significantly low in shifter, multiplier and divider, as compared to DCT. When the *short-sleep* prediction misses, PG is disabled in spite of the fact that the actual sleep is *long-sleep*. This loses opportunities for energy savings at *long-sleep*. AHB-LM described in Section 3.4 solves this problem. The loss in energy savings due to mis-prediction is minimized by using AHB-LM. As a result, AHB-LM saves more energy than ATB by 4% at 65°C and by 3% at 100°C at QSORT.

AHB-LM achieved the best energy savings among all in this study. As compared to non-PG, power gating based on the AHB-LM policy reduces the total energy dissipation of four function units to 41.4% for DCT and to 11.8% for QSORT at 100°C. We also compared with the *Oracle* case in which the prediction always hits at every sleep event. Results showed that energy is saved to 39.8% for DCT and to 11.8% for QSORT at 100°C in *Oracle*. Thus, energy savings achieved by AHB-LM are very close to those of *Oracle*.

Notice that performance overhead is not introduced by employing our adaptive and history based approaches. As described in Section 4.1, the function units are woken up by a

pre-wakeup mechanism at the IF stage. This mechanism works independently of the decision on whether a unit is put to sleep or not. As a result, the wakeup time is not changed and hence the wakeup latency is concealed in the pipeline.

6. Conclusions

We described an adaptive power-gating approach based on temperature dependent break-even time (BET) for function units. An analytical model to express the temperature dependent BET was presented and the accuracy of the model was examined using real data of function units. We proposed four power-gating policies in this paper. Among them, an adaptive history-based policy with a limiter demonstrated the best energy savings.

As a future work, we will study a design technique to reduce transient-glitch energy for further energy savings.

Appendix

In the analytical model for BET proposed by Hu *et al* [3], temperature dependence is not considered. We re-examined the procedure of how the model was built and checked if the derivation steps still work at high temperature. Under an assumption that a pMOS header PS is used, they formulated the voltage drop at the virtual-V_{dd} (VVDD) line. The voltage drop at VVDD is caused by the fact that electric charge at the VVDD line leaks to the ground after PS is turned off. Suppose that while the circuit is power gated, the voltage V at the VVDD line drops by ΔV_{cyc}^{i} during cycle *i*. Hu *et al* assumed that for the first cycle the leakage current approximately equals to the leakage current through the circuit while it is active, because the voltage at the VVDD line changes by only a small amount in one cycle. This assumption may be valid at the room temperature. However, as the temperature goes up, leakage increases and thereby *the voltage at the VVDD line could change so much even in one cycle at the high temperature*.

Furthermore, the following expression was derived for the voltage drop in subsequent cycle *i*:

$$\Delta V_{cyc}^{i} = \Delta V_{cyc}^{0} \exp(\frac{\eta}{mU_t}(V - V_{dd})). \quad \text{(A-1)}$$

By using the fact that the argument of the exponent in expression (A-1) is smaller enough than 1, the authors applied a linear approximation $\exp(-x) \approx 1 - x$. However, it should be noted that this approximation works *only when the voltage drop at the VVDD line during one cycle is small enough*. This condition may be satisfied at the room temperature but is not necessarily guaranteed at the high temperature.

To cope with this problem, we divide the period of one cycle into n' small pieces. Then the time period for a piece is expressed as $\Delta t = T_{cyc} / n'$. Suppose that the voltage at the VVDD line drops by $\Delta V_{\Delta t}^{i}$ during the *i*-th Δt period while the circuit is power gated. Notice that for the first Δt period the leakage current approximately equals to the leakage current through the circuit while it is active, because the voltage at the VVDD line changes by only a small amount during Δt period. This assumption becomes valid by choosing a large enough *n'*. The average leakage energy dissipated per Δt period when the circuit is active is $E_{\Delta t}^{L} = E_{cyc}^{L} / n'$. The leakage energy dissipated during the first Δt period can be expressed as

$$E_{\Delta t}^{0} = E_{\Delta t}^{L} = E_{cyc}^{L} / n'. \quad \text{(A-2)}$$

For the first Δt period that the circuit is power gated, the amount of energy needed to restore the voltage at the VVDD line to the V_{dd} level is $E_{\Delta t}^{0} = (C_D + (1/2)C_S)V_{dd}\Delta V_{\Delta t}^{0}$, where C_D is the total capacitance at the VVDD line; C_S is the total switching capacitance in the circuit; and $\Delta V_{\Delta t}^{0}$ is the voltage drop at VVDD during the first Δt period. Then we obtain

$$\Delta V_{\Delta t}^{0} = \frac{1}{n'} \bullet \frac{1}{2} \alpha L V_{dd} (\frac{1}{2} + \frac{C_D}{C_S})^{-1} \quad \text{(A-3)}$$

where L is the leakage factor described in Section 2. The voltage drop at the VVDD line during the *i*-th Δt period can be expressed as

$$\Delta V_{\Delta t}^{i} = \Delta V_{\Delta t}^{0} \exp(\frac{\eta}{mU_t}(V - V_{dd})). \quad \text{(A-4)}$$

Δt can be chosen small enough such that the argument of the exponent in expression (A-4) becomes smaller enough than 1. This allows us to use a linear expression for the exponent. Then we obtain

$$\Delta V_{\Delta t}^{i} \approx \Delta V_{\Delta t}^{0} (1 - i\frac{\eta}{mU_t}\Delta V_{\Delta t}^{0}). \quad \text{(A-5)}$$

Energy savings during the *i*-th Δt period can be expressed as

$$E_{\Delta t\_saved}^{i} = E_{\Delta t}^{L} - E_{\Delta t}^{i} = E_{\Delta t}^{L} i\frac{\eta}{mU_t}\Delta V_{\Delta t}^{0}. \quad \text{(A-6)}$$

The total (aggregate) energy saved over N cycle period (i.e. $Nn'\Delta t$) can be expressed as

$$E_{ag\_saved}^{N} = \sum_{i=0}^{n'N} E_{\Delta t\_saved}^{i} = E_{\Delta t}^{L} \frac{\eta}{mU_t}\Delta V_{\Delta t}^{0} \frac{n'^2 N^2}{2} \quad . \quad \text{(A-7)}$$

Using expressions (A-2), (A-3) and (A-7), we finally obtain

$$E_{ag\_saved}^{N} = E_{cyc}^{L} \frac{\eta}{mU_t} \frac{N^2}{2} \frac{\alpha L V_{dd}}{2} (\frac{1}{2} + \frac{C_D}{C_S})^{-1}. \quad \text{(A-8)}$$

This ended up with an expression that is *not* dependent on n'. Also, expression (A-8) is exactly same as the one derived by Hu *et al* [3]. From (A-8) and expression for energy overhead for driving PS, we obtain expression (1) in Section 2 for BEC.

Acknowledgment

The authors thank VLSI Design and Education Center (VDEC), Synopsys, Cadence, Mentor, Apache Design Solutions, STARC, and Japan Science and Technology Agency (JST) CREST for their support.

References

[1] T. Hattori, et. al., "Hierarchical power distribution and power management scheme for a single chip mobile processor," *Proc. Design Automation Conference*, pp.292-295, 2006.

[2] R. Kumar and G. Hinton, "A family of 45nm IA processors," *ISSCC'09 Digest of Technical Papers*, pp.58-59, Feb. 2009.

[3] Z. Hu, et al, "Microarchitectural techniques for power gating of execution units," *Proc. ISLPED'04*, pp.32-37, 2004.

[4] N. Seki, et al, "A Fine-grain Dynamic Sleep Control Scheme in MIPS R3000", *Proc. ICCD'08*, Oct. 2008.

[5] K. Usami and N. Ohkubo, "A design approach for fine-grained run-time power gating using locally extracted sleep signals," *Proc. ICCD'06*, pp.155-161, Oct. 2006.

[6] H. Matsutani, et al, "Run-Time Power Gating of On-Chip Routers Using Look-Ahead Routing," *Proc. ASP-DAC'08*, pp.55-60, Jan. 2008.

[7] K. Usami et al, "Design and implementation of fine-grain power gating with ground bounce suppression," *Proc. Int. Conf. on VLSI Design ' 09*, pp. 381-386, Jan. 2009.

[8] S. Rele, et al, "Optimizing static power dissipation by functional units in superscalar processors," *Computational Complexity*, pp. 261–275, 2002.

[9] A. Youssef, et al, "Dynamic standby prediction for leakage tolerant microprocessor functional units," *Proc. Int. Symp. on Microarchitecture*, Dec. 2006.

[10] K. Usami, et al, "Implementation and evaluation of fine-grain run-time power gating for a multiplier," *Proc. ICICDT '09*, pp. 7-10, May 2009.

[11] D. Kannan, et al, "Temperature and process variations aware power gating of functional units," *Proc. Int. Conf. on VLSI Design '08*, pp. 515-520, Jan. 2008.

[12] J. Clabes, et al, "Design and implementation of the POWER5™ microprocessor," *ISSCC'04 Digest of Technical Papers*, Feb. 2004.

[13] Taur and Ning, *Fundamentals of modern VLSI devices*, Cambridge University Press, p.120, 1998.

[14] J. Frenkil and S. Venkatraman, "Power gating design automation," in *Closing the Power Gap Between ASIC & Custom*, ed. by D. Chinnery and K. Keutzer, Springer, pp. 269-272, 2007.